Programming Exercise 12
Math Games

C# Step by Step

This exercise consists of building a mathematics flash card simulator. You will have four "sets" of flash cards, addition, subtraction, multiplication, and division. To initialize a session, you will ask the users to select the type of problem they want too practice and the number of problems. You will echo the problem type and the number of problems to the user. See figure 1.

When the user begins the test, the application will present the specified number of problems of the specified type. The problems will generate random integers between 1 and 12, or whatever range you choose. The application will keep score, and if the user misses a problem the application will display the correct answer to the problem. At the conclusion of the session the application will report the number of correctly answered questions and give a numeric score. See figure 2.

To help you get started, I have suggested some started code below. If you use this starter code (you do not have to, you can do this any way you choose), you will need to implement six methods: `Initialize()`, `Add(n)`, `Subtract(n)`, `Multiply(n)`, `Divide(n)`, and `Report(n1, n2)`. All of these methods will return values.

Please note the following: (1) I have written the `Subtract()` method so that the answer is always a positive number, that is, the larger integer is on the left hand side of the operator and the smaller integer is on the right hand side of the operator. (2) The `Divide()` method will return a floating point number, and you will have to use an approximate answer to check for correctness. For example $\frac{3}{7} = 0.428571428571428571428571428571428571$, but you would probably accept 0.42 or 0.43 as correct.

Figure 1: Math Games Configuration

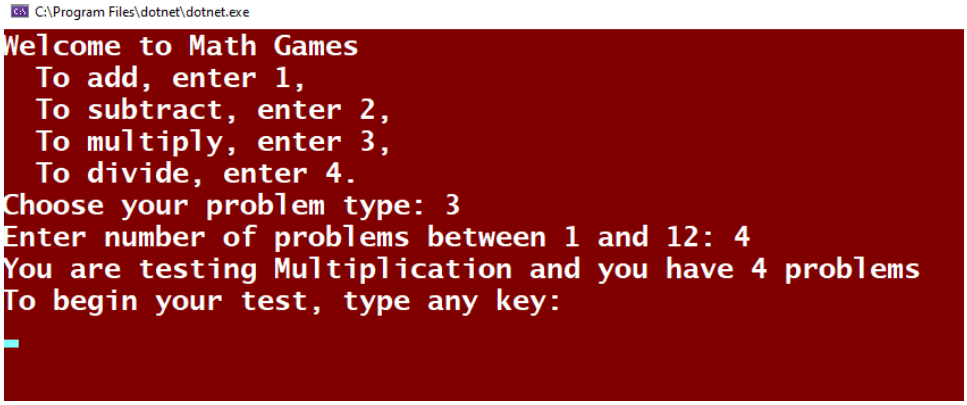
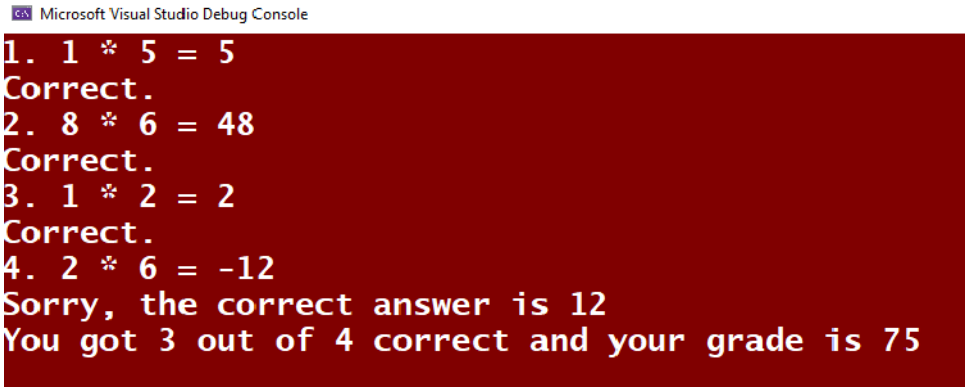


Figure 2: Math Games Problems




```
1  namespace MathGames
2  {
3      class Program
4      {
5          static void Main(string[] args)
6          {
7              Console.WriteLine("Welcome to Math Games");
8              int probType = 0;
9              int numProb = 0;
10             int score = 0;
11             (probType, numProb) = Util.Initialize();
12             if (probType == 1)
13                 score = Util.Add(numProb);
14             else if (probType == 2)
15                 score = Util.Subtract(numProb);
16             else if (probType == 3)
17                 score = Util.Multiply(numProb);
18             else if (probType == 4)
19                 score = Util.Divide(numProb);
20             else
21                 Console.WriteLine("Sorry, you made an invalid choice.");
22             string report = Util.Report(score, numProb);
23             Console.WriteLine(report);
24         }
25     }
26 }
```